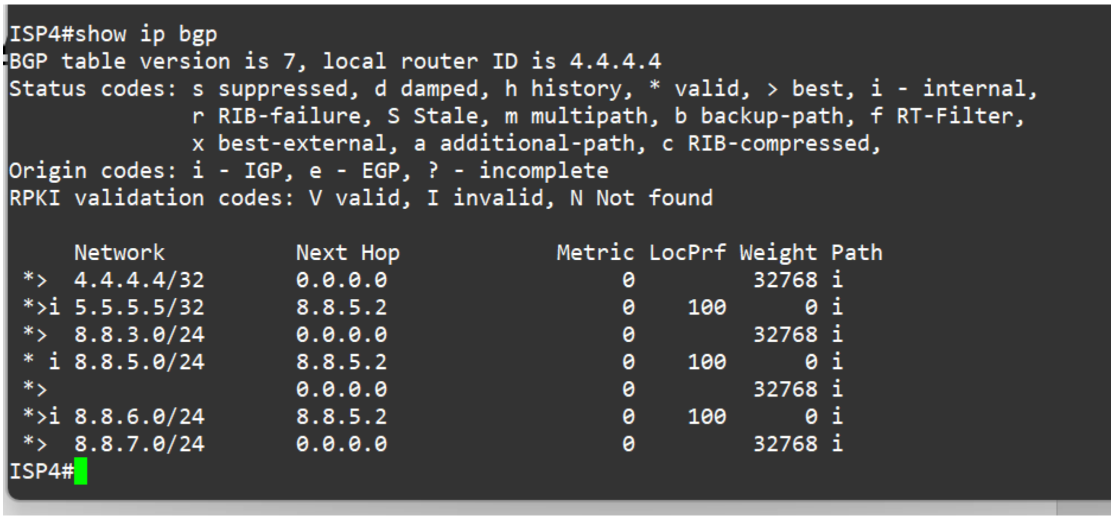
DMVPN LAB-report

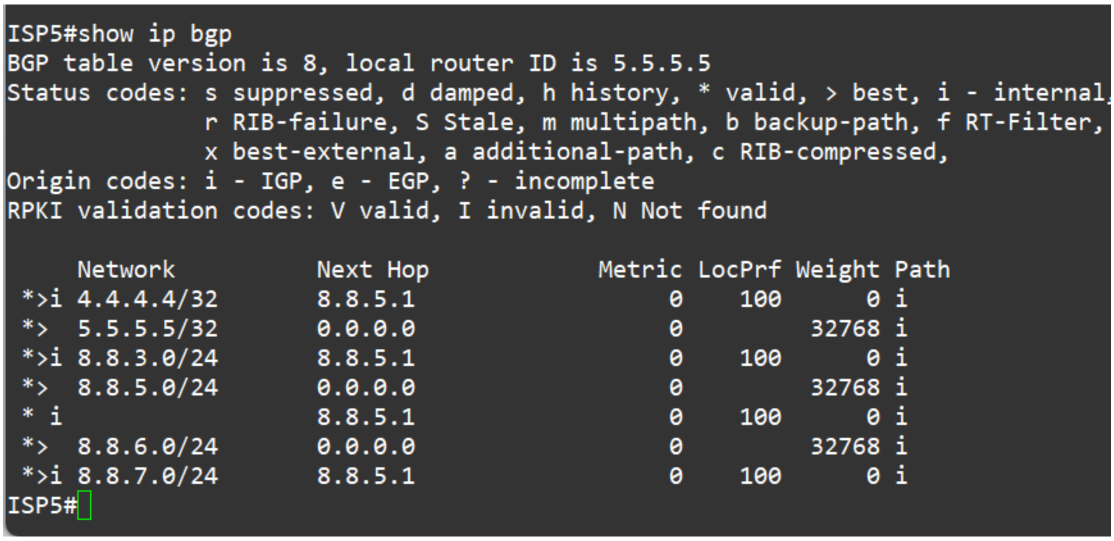
BGP

5.b. ISP4:



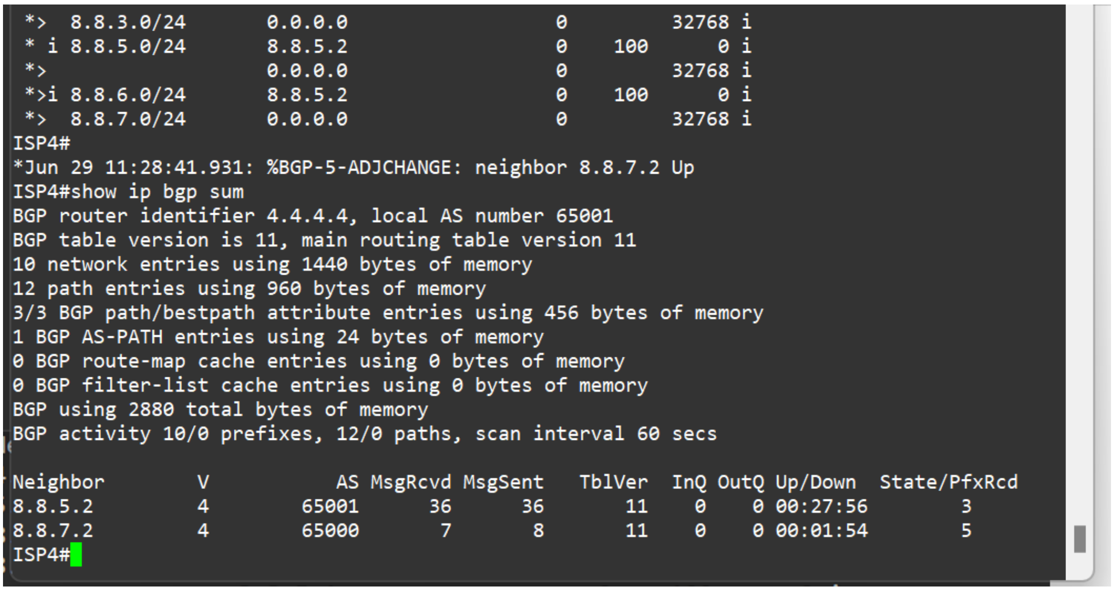
```
ISP4#show ip bgp
BGP table version is 7, local router ID is 4.4.4.4
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>  4.4.4.4/32       0.0.0.0                  0         32768 i
 *>i 5.5.5.5/32       8.8.5.2                  0    100      0 i
 *>  8.8.3.0/24       0.0.0.0                  0         32768 i
 * i 8.8.5.0/24       8.8.5.2                  0    100      0 i
 *>                   0.0.0.0                  0         32768 i
 *>i 8.8.6.0/24       8.8.5.2                  0    100      0 i
 *>  8.8.7.0/24       0.0.0.0                  0         32768 i
ISP4#
```

ISP5:



```
ISP5#show ip bgp
BGP table version is 8, local router ID is 5.5.5.5
Status codes: s suppressed, d damped, h history, * valid, > best, i - internal,
              r RIB-failure, S Stale, m multipath, b backup-path, f RT-Filter,
              x best-external, a additional-path, c RIB-compressed,
Origin codes: i - IGP, e - EGP, ? - incomplete
RPKI validation codes: V valid, I invalid, N Not found

     Network          Next Hop            Metric LocPrf Weight Path
 *>i 4.4.4.4/32       8.8.5.1                  0    100      0 i
 *>  5.5.5.5/32       0.0.0.0                  0         32768 i
 *>i 8.8.3.0/24       8.8.5.1                  0    100      0 i
 *>  8.8.5.0/24       0.0.0.0                  0         32768 i
 * i                  8.8.5.1                  0    100      0 i
 *>  8.8.6.0/24       0.0.0.0                  0         32768 i
 *>i 8.8.7.0/24       8.8.5.1                  0    100      0 i
ISP5#
```

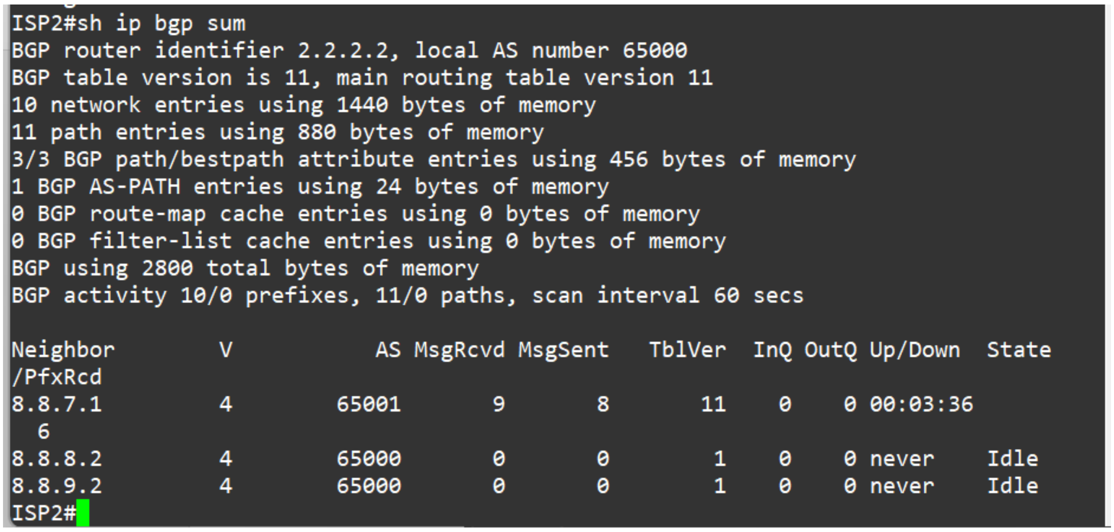
c.



```
 *>  8.8.3.0/24       0.0.0.0                  0         32768 i
 * i 8.8.5.0/24       8.8.5.2                  0    100      0 i
 *>                   0.0.0.0                  0         32768 i
 *>i 8.8.6.0/24       8.8.5.2                  0    100      0 i
 *>  8.8.7.0/24       0.0.0.0                  0         32768 i
ISP4#
*Jun 29 11:28:41.931: %BGP-5-ADJCHANGE: neighbor 8.8.7.2 Up
ISP4#show ip bgp sum
BGP router identifier 4.4.4.4, local AS number 65001
BGP table version is 11, main routing table version 11
10 network entries using 1440 bytes of memory
12 path entries using 960 bytes of memory
3/3 BGP path/bestpath attribute entries using 456 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2880 total bytes of memory
BGP activity 10/0 prefixes, 12/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State/PfxRcd
8.8.5.2         4        65001      36      36       11    0    0 00:27:56        3
8.8.7.2         4        65000       7       8       11    0    0 00:01:54        5
ISP4#
```

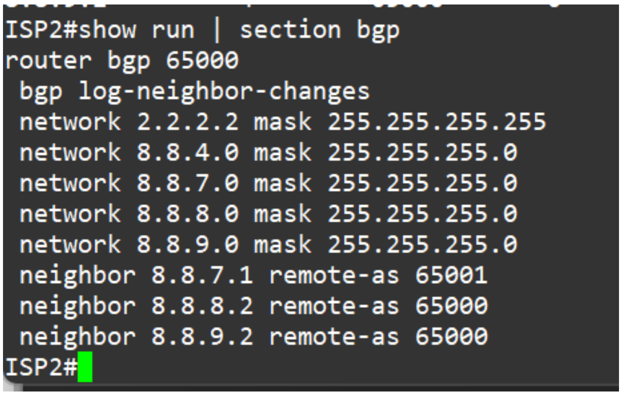
The local router learned 3 routes or pfx through neighbor 8.8.5.2 in AS 65001. And the local router learned 5 routes or pfx through neighbor 8.8.7.2 in AS 65000

D.



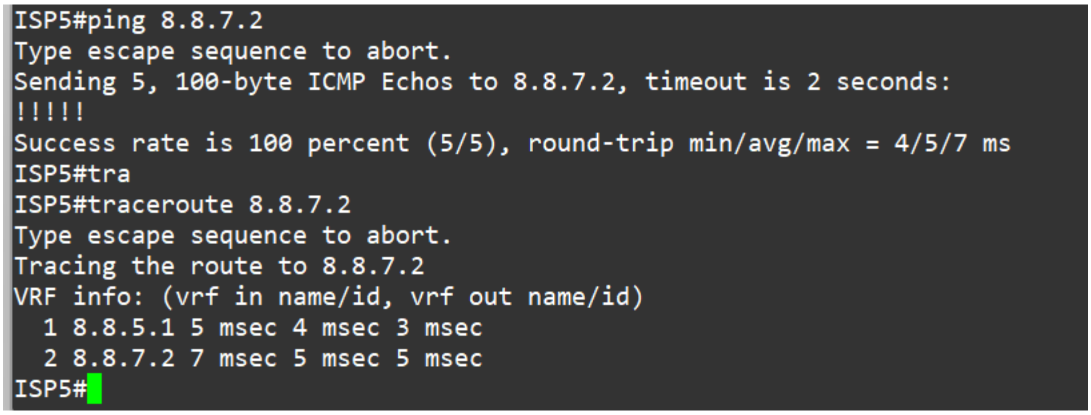
```
ISP2#sh ip bgp sum
BGP router identifier 2.2.2.2, local AS number 65000
BGP table version is 11, main routing table version 11
10 network entries using 1440 bytes of memory
11 path entries using 880 bytes of memory
3/3 BGP path/bestpath attribute entries using 456 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2800 total bytes of memory
BGP activity 10/0 prefixes, 11/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State
/PfxRcd
8.8.7.1         4        65001       9       8       11    0    0 00:03:36
  6
8.8.8.2         4        65000       0       0        1    0    0 never    Idle
8.8.9.2         4        65000       0       0        1    0    0 never    Idle
ISP2#
```

e.



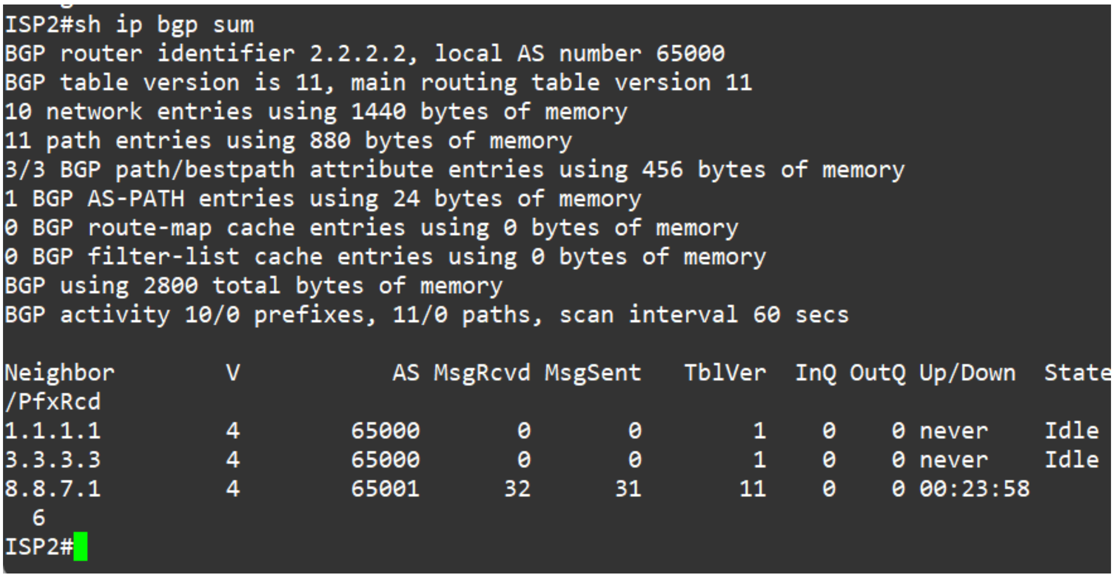
```
ISP2#show run | section bgp
router bgp 65000
 bgp log-neighbor-changes
 network 2.2.2.2 mask 255.255.255.255
 network 8.8.4.0 mask 255.255.255.0
 network 8.8.7.0 mask 255.255.255.0
 network 8.8.8.0 mask 255.255.255.0
 network 8.8.9.0 mask 255.255.255.0
 neighbor 8.8.7.1 remote-as 65001
 neighbor 8.8.8.2 remote-as 65000
 neighbor 8.8.9.2 remote-as 65000
ISP2#
```

g.



```
ISP5#ping 8.8.7.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.7.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/5/7 ms
ISP5#tra
ISP5#traceroute 8.8.7.2
Type escape sequence to abort.
Tracing the route to 8.8.7.2
VRF info: (vrf in name/id, vrf out name/id)
  1 8.8.5.1 5 msec 4 msec 3 msec
  2 8.8.7.2 7 msec 5 msec 5 msec
ISP5#
```

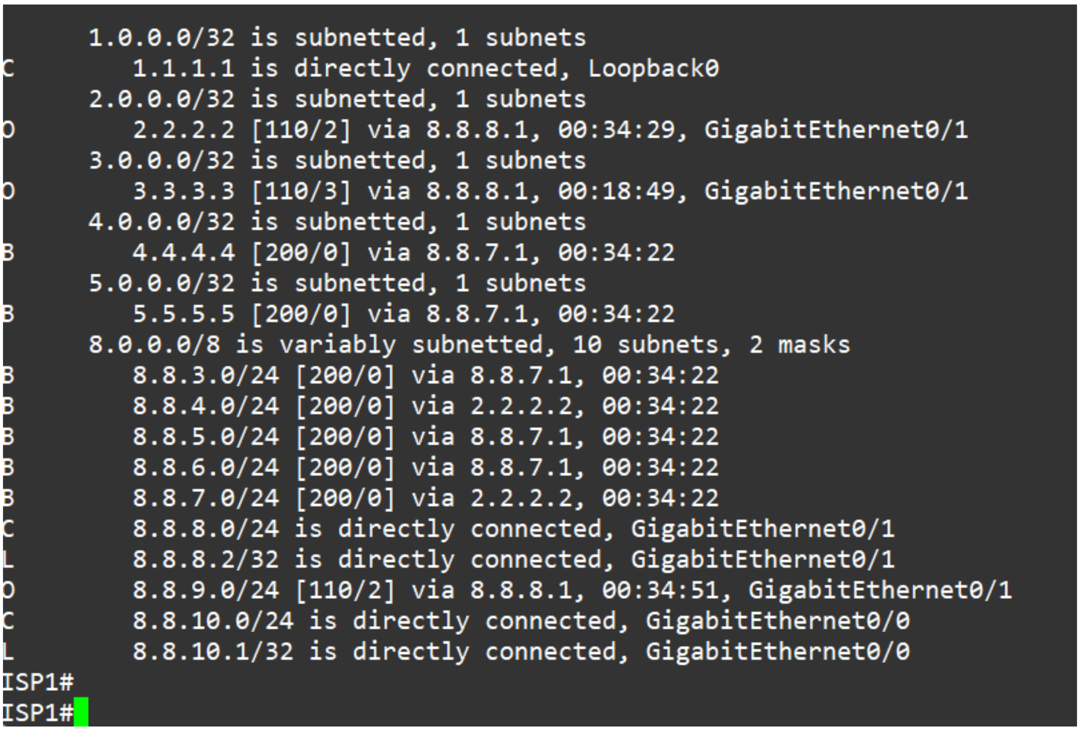
i.



```
ISP2#sh ip bgp sum
BGP router identifier 2.2.2.2, local AS number 65000
BGP table version is 11, main routing table version 11
10 network entries using 1440 bytes of memory
11 path entries using 880 bytes of memory
3/3 BGP path/bestpath attribute entries using 456 bytes of memory
1 BGP AS-PATH entries using 24 bytes of memory
0 BGP route-map cache entries using 0 bytes of memory
0 BGP filter-list cache entries using 0 bytes of memory
BGP using 2800 total bytes of memory
BGP activity 10/0 prefixes, 11/0 paths, scan interval 60 secs

Neighbor        V           AS MsgRcvd MsgSent   TblVer  InQ OutQ Up/Down  State
/PfxRcd
1.1.1.1         4        65000       0       0        1    0    0 never    Idle
3.3.3.3         4        65000       0       0        1    0    0 never    Idle
8.8.7.1         4        65001      32      31       11    0    0 00:23:58
  6
ISP2#
```

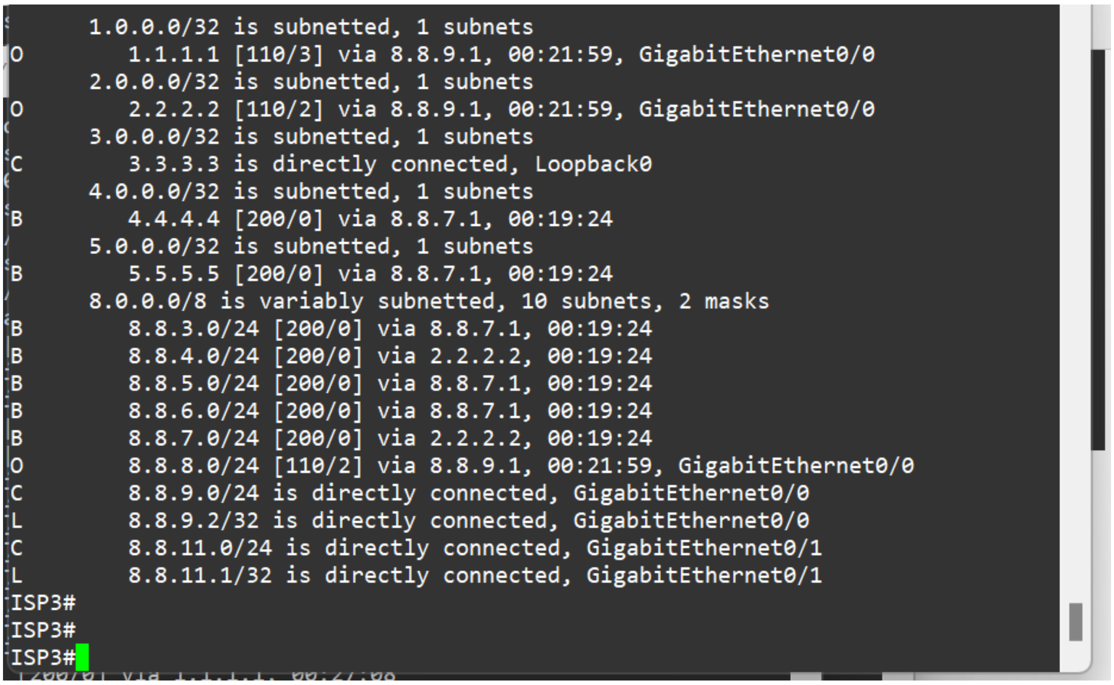
IGP - OSPF

k. ISP 1:



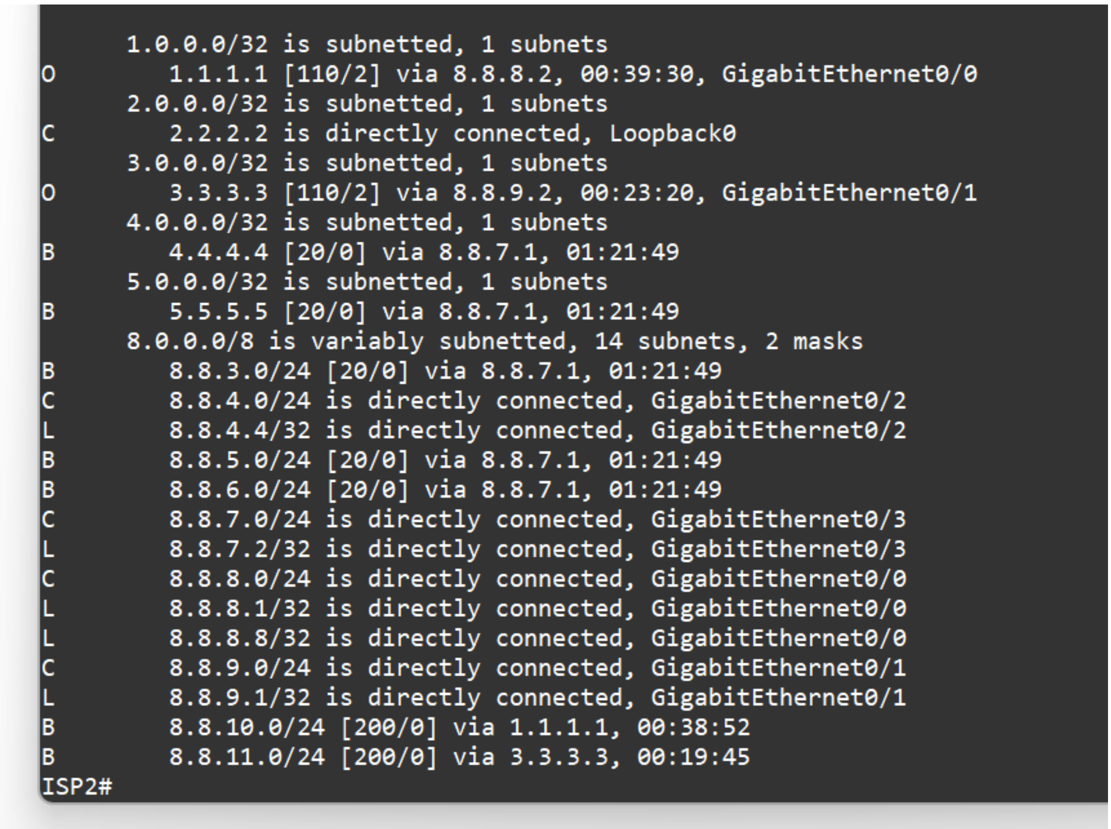
```
      1.0.0.0/32 is subnetted, 1 subnets
C        1.1.1.1 is directly connected, Loopback0
      2.0.0.0/32 is subnetted, 1 subnets
O        2.2.2.2 [110/2] via 8.8.8.1, 00:34:29, GigabitEthernet0/1
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/3] via 8.8.8.1, 00:18:49, GigabitEthernet0/1
      4.0.0.0/32 is subnetted, 1 subnets
B        4.4.4.4 [200/0] via 8.8.7.1, 00:34:22
      5.0.0.0/32 is subnetted, 1 subnets
B        5.5.5.5 [200/0] via 8.8.7.1, 00:34:22
      8.0.0.0/8 is variably subnetted, 10 subnets, 2 masks
B        8.8.3.0/24 [200/0] via 8.8.7.1, 00:34:22
B        8.8.4.0/24 [200/0] via 2.2.2.2, 00:34:22
B        8.8.5.0/24 [200/0] via 8.8.7.1, 00:34:22
B        8.8.6.0/24 [200/0] via 8.8.7.1, 00:34:22
B        8.8.7.0/24 [200/0] via 2.2.2.2, 00:34:22
C        8.8.8.0/24 is directly connected, GigabitEthernet0/1
L        8.8.8.2/32 is directly connected, GigabitEthernet0/1
O        8.8.9.0/24 [110/2] via 8.8.8.1, 00:34:51, GigabitEthernet0/1
C        8.8.10.0/24 is directly connected, GigabitEthernet0/0
L        8.8.10.1/32 is directly connected, GigabitEthernet0/0
ISP1#
ISP1#
```

ISP3:



```
      1.0.0.0/32 is subnetted, 1 subnets
O        1.1.1.1 [110/3] via 8.8.9.1, 00:21:59, GigabitEthernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
O        2.2.2.2 [110/2] via 8.8.9.1, 00:21:59, GigabitEthernet0/0
      3.0.0.0/32 is subnetted, 1 subnets
C        3.3.3.3 is directly connected, Loopback0
      4.0.0.0/32 is subnetted, 1 subnets
B        4.4.4.4 [200/0] via 8.8.7.1, 00:19:24
      5.0.0.0/32 is subnetted, 1 subnets
B        5.5.5.5 [200/0] via 8.8.7.1, 00:19:24
      8.0.0.0/8 is variably subnetted, 10 subnets, 2 masks
B        8.8.3.0/24 [200/0] via 8.8.7.1, 00:19:24
B        8.8.4.0/24 [200/0] via 2.2.2.2, 00:19:24
B        8.8.5.0/24 [200/0] via 8.8.7.1, 00:19:24
B        8.8.6.0/24 [200/0] via 8.8.7.1, 00:19:24
B        8.8.7.0/24 [200/0] via 2.2.2.2, 00:19:24
O        8.8.8.0/24 [110/2] via 8.8.9.1, 00:21:59, GigabitEthernet0/0
C        8.8.9.0/24 is directly connected, GigabitEthernet0/0
L        8.8.9.2/32 is directly connected, GigabitEthernet0/0
C        8.8.11.0/24 is directly connected, GigabitEthernet0/1
L        8.8.11.1/32 is directly connected, GigabitEthernet0/1
ISP3#
ISP3#
ISP3#
 [200/0] via 1.1.1.1, 00:27:08
```

i. ISP2:

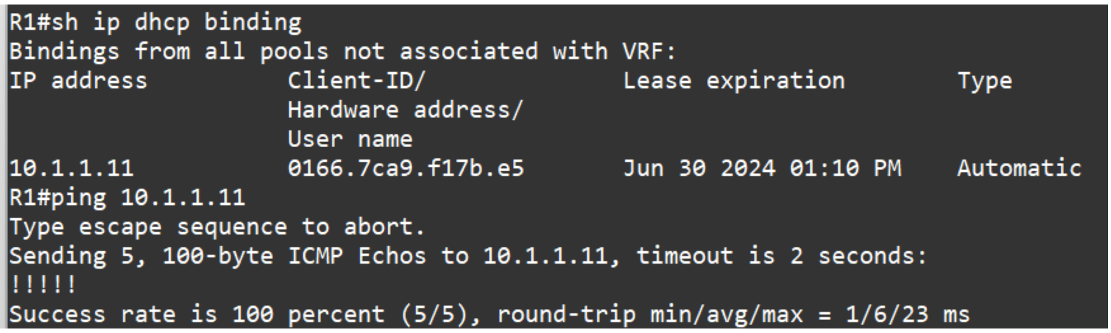


```
      1.0.0.0/32 is subnetted, 1 subnets
O        1.1.1.1 [110/2] via 8.8.8.2, 00:39:30, GigabitEthernet0/0
      2.0.0.0/32 is subnetted, 1 subnets
C        2.2.2.2 is directly connected, Loopback0
      3.0.0.0/32 is subnetted, 1 subnets
O        3.3.3.3 [110/2] via 8.8.9.2, 00:23:20, GigabitEthernet0/1
      4.0.0.0/32 is subnetted, 1 subnets
B        4.4.4.4 [20/0] via 8.8.7.1, 01:21:49
      5.0.0.0/32 is subnetted, 1 subnets
B        5.5.5.5 [20/0] via 8.8.7.1, 01:21:49
      8.0.0.0/8 is variably subnetted, 14 subnets, 2 masks
B        8.8.3.0/24 [20/0] via 8.8.7.1, 01:21:49
C        8.8.4.0/24 is directly connected, GigabitEthernet0/2
L        8.8.4.4/32 is directly connected, GigabitEthernet0/2
B        8.8.5.0/24 [20/0] via 8.8.7.1, 01:21:49
B        8.8.6.0/24 [20/0] via 8.8.7.1, 01:21:49
C        8.8.7.0/24 is directly connected, GigabitEthernet0/3
L        8.8.7.2/32 is directly connected, GigabitEthernet0/3
C        8.8.8.0/24 is directly connected, GigabitEthernet0/0
L        8.8.8.1/32 is directly connected, GigabitEthernet0/0
L        8.8.8.8/32 is directly connected, GigabitEthernet0/0
C        8.8.9.0/24 is directly connected, GigabitEthernet0/1
L        8.8.9.1/32 is directly connected, GigabitEthernet0/1
B        8.8.10.0/24 [200/0] via 1.1.1.1, 00:38:52
B        8.8.11.0/24 [200/0] via 3.3.3.3, 00:19:45
ISP2#
```

m. this is required because we haven’t configured ISP2(part3 ~"2:40")

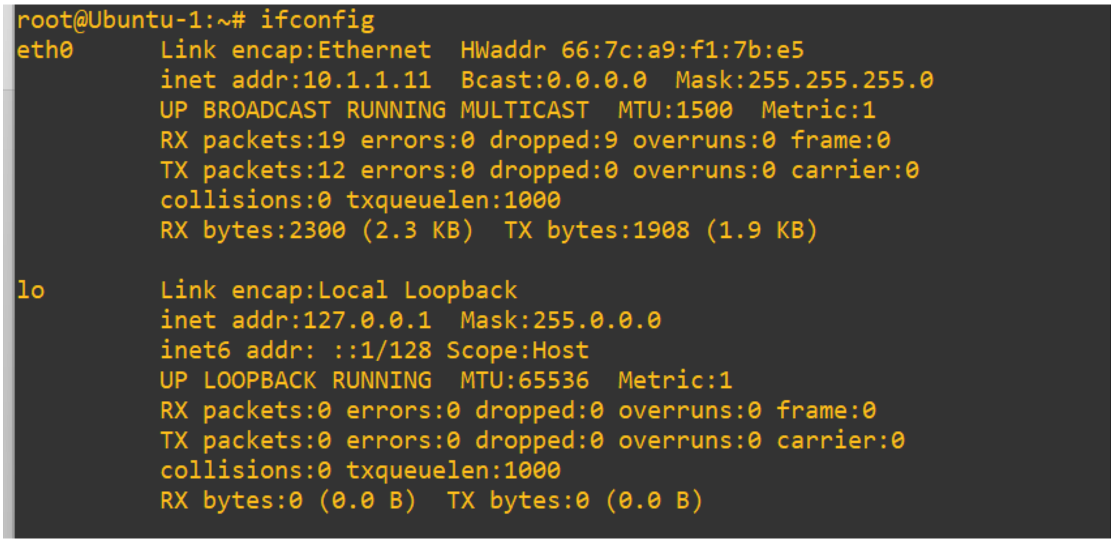
6.DHCP pools on the ubuntu clients

e-f.



```
R1#sh ip dhcp binding
Bindings from all pools not associated with VRF:
IP address          Client-ID/              Lease expiration        Type
                    Hardware address/
                    User name
10.1.1.11           0166.7ca9.f17b.e5       Jun 30 2024 01:10 PM    Automatic
R1#ping 10.1.1.11
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.1.11, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/6/23 ms
```

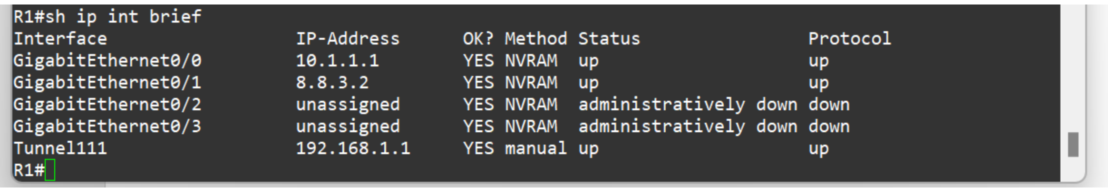
g.



```
root@Ubuntu-1:~# ifconfig
eth0      Link encap:Ethernet  HWaddr 66:7c:a9:f1:7b:e5
          inet addr:10.1.1.11  Bcast:0.0.0.0  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:19 errors:0 dropped:9 overruns:0 frame:0
          TX packets:12 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:2300 (2.3 KB)  TX bytes:1908 (1.9 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:0 (0.0 B)  TX bytes:0 (0.0 B)
```

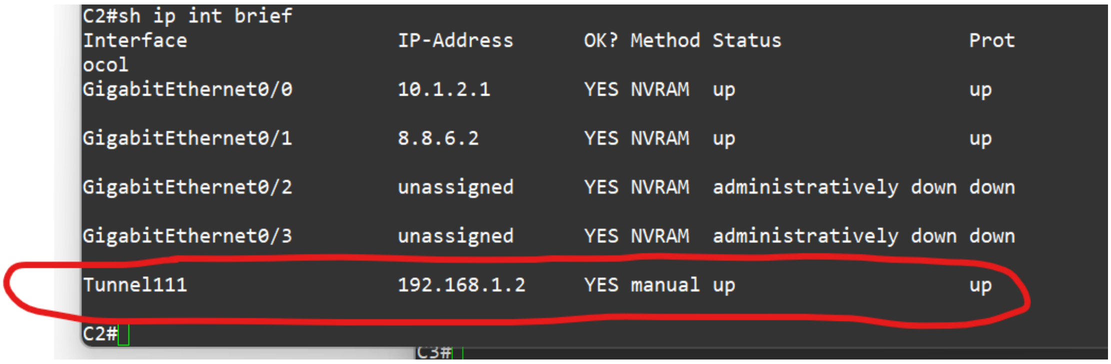
7. DMVPN (Using EIGRP, mGRE and NHRP):

c. C1:



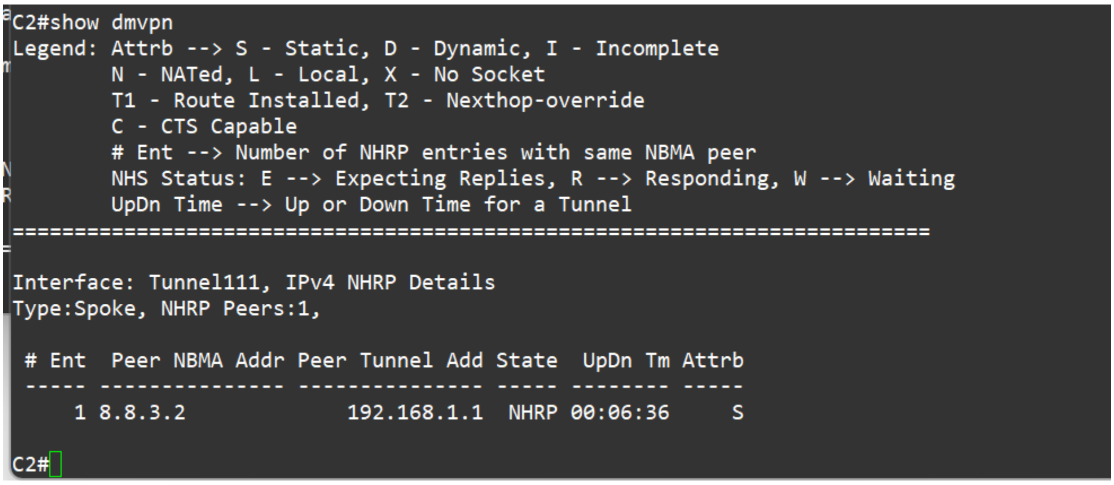
```
R1#sh ip int brief
Interface                  IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0         10.1.1.1        YES NVRAM  up                    up
GigabitEthernet0/1         8.8.3.2         YES NVRAM  up                    up
GigabitEthernet0/2         unassigned      YES NVRAM  administratively down down
GigabitEthernet0/3         unassigned      YES NVRAM  administratively down down
Tunnel111                  192.168.1.1     YES manual up                    up
R1#
```

C2:



```
C2#sh ip int brief
Interface                  IP-Address      OK? Method Status                Prot
ocol
GigabitEthernet0/0         10.1.2.1        YES NVRAM  up                    up

GigabitEthernet0/1         8.8.6.2         YES NVRAM  up                    up

GigabitEthernet0/2         unassigned      YES NVRAM  administratively down down

GigabitEthernet0/3         unassigned      YES NVRAM  administratively down down

Tunnel111                  192.168.1.2     YES manual up                    up

C2#
                       C3#
```

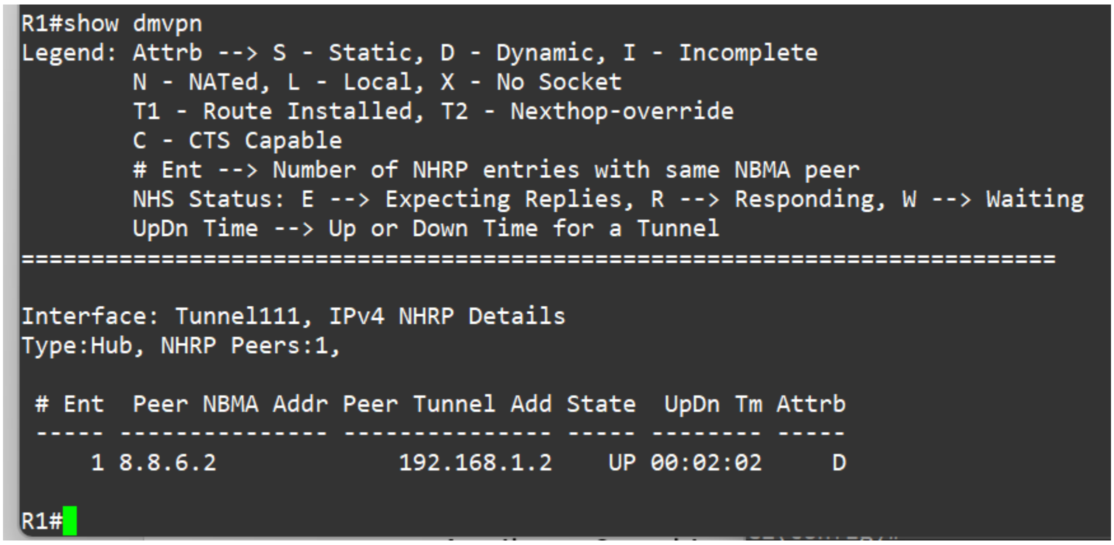
f. C2:



```
C2#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel111, IPv4 NHRP Details
Type:Spoke, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 8.8.3.2             192.168.1.1  NHRP 00:06:36     S

C2#
```

Attribute S and it mean Static

C1:

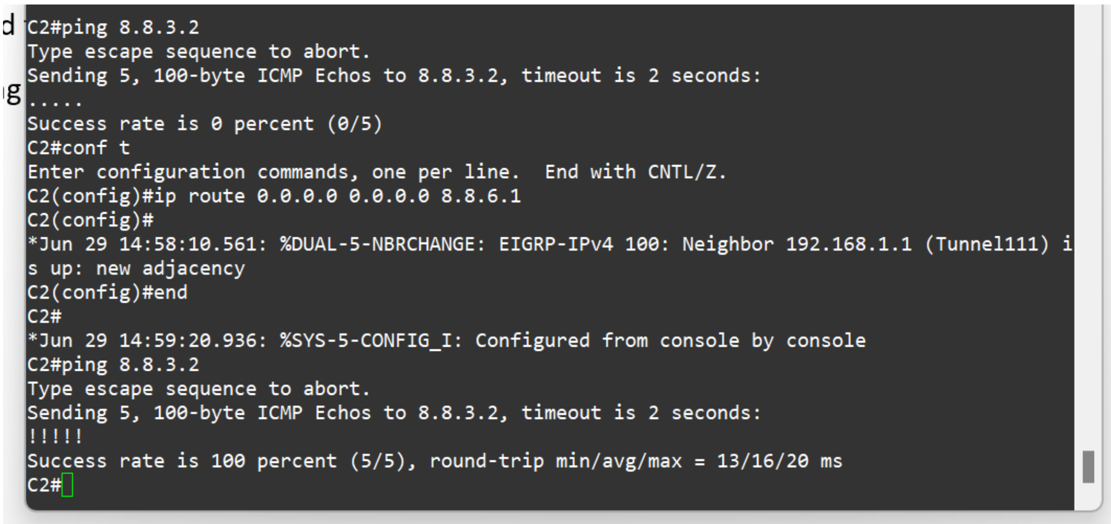


```
R1#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel111, IPv4 NHRP Details
Type:Hub, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 8.8.6.2             192.168.1.2    UP 00:02:02     D

R1#
```

g. no it does not work. And that because we didn’t configure a static default route to my ISP. And I need to do that in both sides.

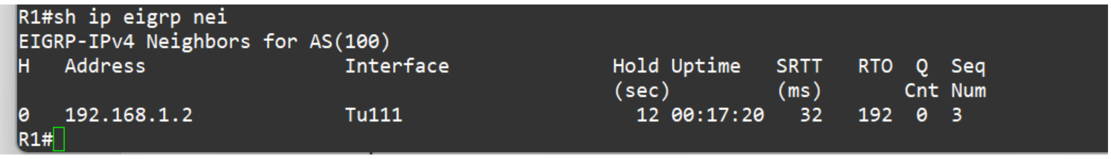
i. yes now the ping is working:

C2:



```
C2#ping 8.8.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.3.2, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
C2#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
C2(config)#ip route 0.0.0.0 0.0.0.0 8.8.6.1
C2(config)#
*Jun 29 14:58:10.561: %DUAL-5-NBRCHANGE: EIGRP-IPv4 100: Neighbor 192.168.1.1 (Tunnel111) i
s up: new adjacency
C2(config)#end
C2#
*Jun 29 14:59:20.936: %SYS-5-CONFIG_I: Configured from console by console
C2#ping 8.8.3.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 8.8.3.2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 13/16/20 ms
C2#
```

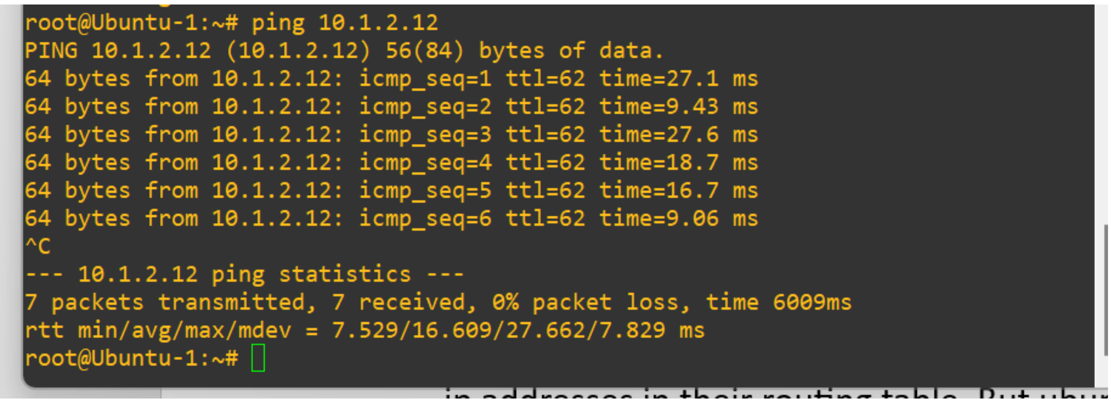
j.



```
R1#sh ip eigrp nei
EIGRP-IPv4 Neighbors for AS(100)
H   Address                 Interface              Hold Uptime   SRTT   RTO  Q  Seq
                                                   (sec)         (ms)       Cnt Num
0   192.168.1.2             Tu111                    12 00:17:20   32   192  0  3
R1#
```

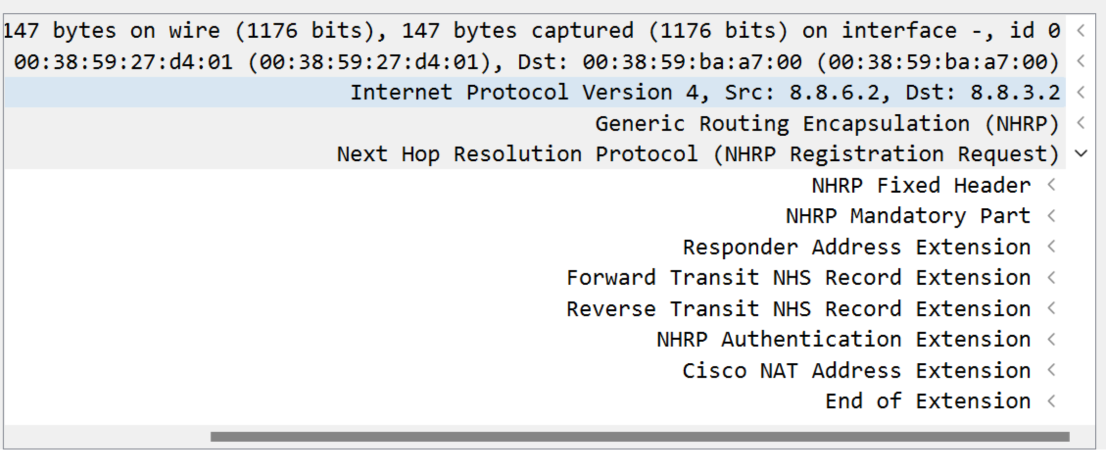
k. the ISP don’t have a visibility of networks 1,2 or network 10. They only have public ip addresses in their routing table. But ubuntu1 and ubuntu2 are able to ping each other across the tunnel and that because the traffic is going throw the Tunnel.

l. yes, the ping is working:



```
root@Ubuntu-1:~# ping 10.1.2.12
PING 10.1.2.12 (10.1.2.12) 56(84) bytes of data.
64 bytes from 10.1.2.12: icmp_seq=1 ttl=62 time=27.1 ms
64 bytes from 10.1.2.12: icmp_seq=2 ttl=62 time=9.43 ms
64 bytes from 10.1.2.12: icmp_seq=3 ttl=62 time=27.6 ms
64 bytes from 10.1.2.12: icmp_seq=4 ttl=62 time=18.7 ms
64 bytes from 10.1.2.12: icmp_seq=5 ttl=62 time=16.7 ms
64 bytes from 10.1.2.12: icmp_seq=6 ttl=62 time=9.06 ms
^C
--- 10.1.2.12 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 6009ms
rtt min/avg/max/mdev = 7.529/16.609/27.662/7.829 ms
root@Ubuntu-1:~#
```

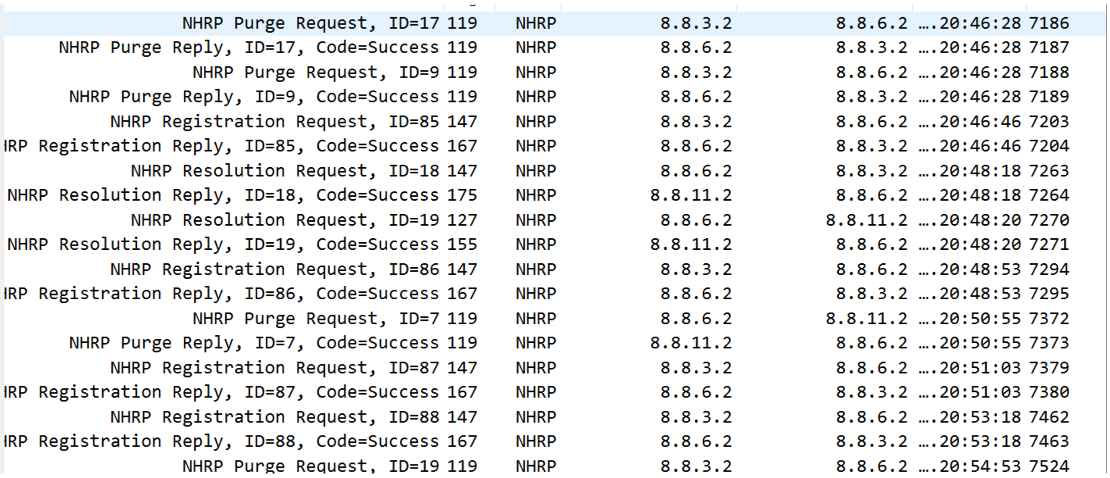
m.



```
147 bytes on wire (1176 bits), 147 bytes captured (1176 bits) on interface -, id 0 <
 00:38:59:27:d4:01 (00:38:59:27:d4:01), Dst: 00:38:59:ba:a7:00 (00:38:59:ba:a7:00) <
                          Internet Protocol Version 4, Src: 8.8.6.2, Dst: 8.8.3.2 <
                                            Generic Routing Encapsulation (NHRP) <
                          Next Hop Resolution Protocol (NHRP Registration Request) v
                                                              NHRP Fixed Header <
                                                            NHRP Mandatory Part <
                                                     Responder Address Extension <
                                               Forward Transit NHS Record Extension <
                                               Reverse Transit NHS Record Extension <
                                                  NHRP Authentication Extension <
                                                     Cisco NAT Address Extension <
                                                              End of Extension <
```

From top to bottom this packet starts off pretty normal, you see the normal L2/L3 information, and the GRE tunnel (since NHRP runs through a GRE tunnel) right after that we get to the NHRP portion of the packet. Now RFC 2332 defines these portions:

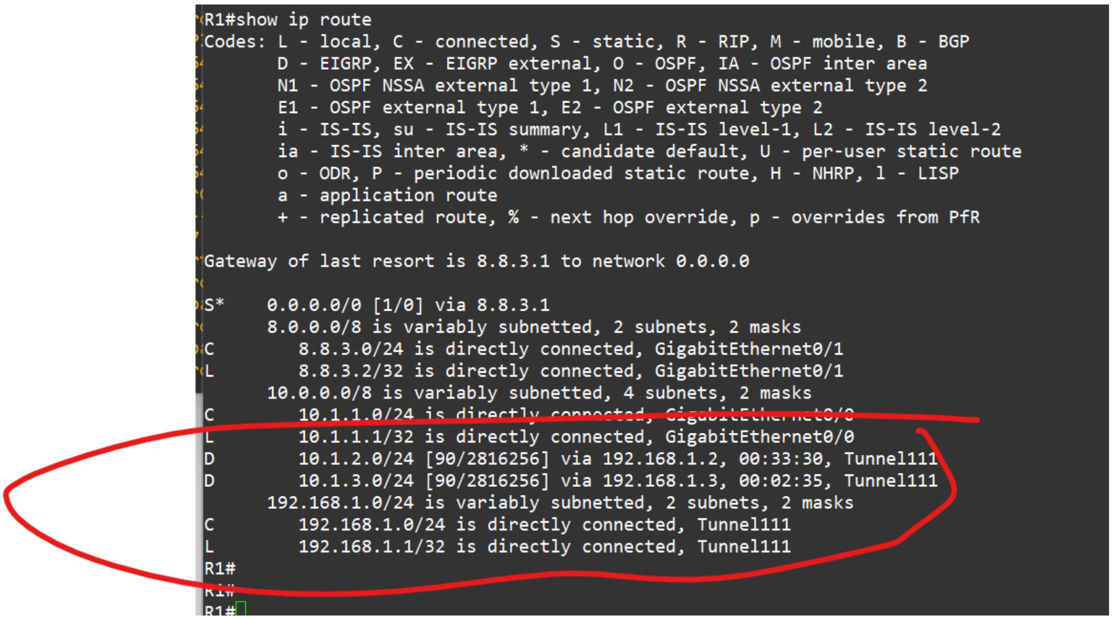
- Fixed Header – This portion is always the same, and just covers some basic information NHRP version, Ethertypes of the physical medium, address family information, and etc.
- Mandatory Part – This is where the NHRP packet type is defined (Registration request/reply, resolution request/reply, etc) and tunnel interface (Displayed as the protocol address) and physical interface (displayed as the NBMA address) IP addresses are contained.
- Client Information Entries CIE: Which contains specific information for clients such as MTU and hold down times.



```
             NHRP Purge Request, ID=17 119   NHRP      8.8.3.2        8.8.6.2 ….20:46:28 7186
   NHRP Purge Reply, ID=17, Code=Success 119   NHRP      8.8.6.2        8.8.3.2 ….20:46:28 7187
              NHRP Purge Request, ID=9 119   NHRP      8.8.3.2        8.8.6.2 ….20:46:28 7188
    NHRP Purge Reply, ID=9, Code=Success 119   NHRP      8.8.6.2        8.8.3.2 ….20:46:28 7189
      NHRP Registration Request, ID=85 147   NHRP      8.8.3.2        8.8.6.2 ….20:46:46 7203
IRP Registration Reply, ID=85, Code=Success 167   NHRP      8.8.6.2        8.8.3.2 ….20:46:46 7204
        NHRP Resolution Request, ID=18 147   NHRP      8.8.6.2        8.8.3.2 ….20:48:18 7263
NHRP Resolution Reply, ID=18, Code=Success 175   NHRP     8.8.11.2        8.8.6.2 ….20:48:18 7264
        NHRP Resolution Request, ID=19 127   NHRP      8.8.6.2       8.8.11.2 ….20:48:20 7270
NHRP Resolution Reply, ID=19, Code=Success 155   NHRP     8.8.11.2        8.8.6.2 ….20:48:20 7271
      NHRP Registration Request, ID=86 147   NHRP      8.8.3.2        8.8.6.2 ….20:48:53 7294
IRP Registration Reply, ID=86, Code=Success 167   NHRP      8.8.6.2        8.8.3.2 ….20:48:53 7295
              NHRP Purge Request, ID=7 119   NHRP      8.8.6.2       8.8.11.2 ….20:50:55 7372
    NHRP Purge Reply, ID=7, Code=Success 119   NHRP     8.8.11.2        8.8.6.2 ….20:50:55 7373
      NHRP Registration Request, ID=87 147   NHRP      8.8.3.2        8.8.6.2 ….20:51:03 7379
IRP Registration Reply, ID=87, Code=Success 167   NHRP      8.8.6.2        8.8.3.2 ….20:51:03 7380
      NHRP Registration Request, ID=88 147   NHRP      8.8.3.2        8.8.6.2 ….20:53:18 7462
IRP Registration Reply, ID=88, Code=Success 167   NHRP      8.8.6.2        8.8.3.2 ….20:53:18 7463
             NHRP Purge Request, ID=19 119   NHRP      8.8.3.2        8.8.6.2 ….20:54:53 7524
```

Above is a quick screenshot of some of the NHRP packet types. You can the NHC at *8.8.3.2* attempts to register with 8.8.6.2 (the NHS) until the NHS sends its registration reply.

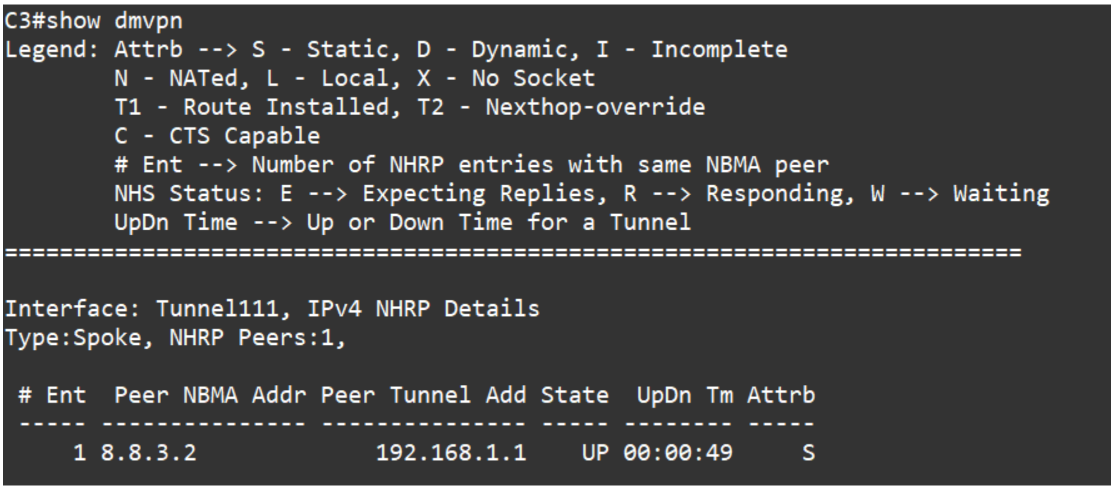
n. yes, as we can see we can see that C2 is connected via a tunnel



```
R1#show ip route
Codes: L - local, C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route, H - NHRP, l - LISP
       a - application route
       + - replicated route, % - next hop override, p - overrides from PfR

Gateway of last resort is 8.8.3.1 to network 0.0.0.0

S*    0.0.0.0/0 [1/0] via 8.8.3.1
      8.0.0.0/8 is variably subnetted, 2 subnets, 2 masks
C        8.8.3.0/24 is directly connected, GigabitEthernet0/1
L        8.8.3.2/32 is directly connected, GigabitEthernet0/1
      10.0.0.0/8 is variably subnetted, 4 subnets, 2 masks
C        10.1.1.0/24 is directly connected, GigabitEthernet0/0
L        10.1.1.1/32 is directly connected, GigabitEthernet0/0
D        10.1.2.0/24 [90/2816256] via 192.168.1.2, 00:33:30, Tunnel111
D        10.1.3.0/24 [90/2816256] via 192.168.1.3, 00:02:35, Tunnel111
      192.168.1.0/24 is variably subnetted, 2 subnets, 2 masks
C        192.168.1.0/24 is directly connected, Tunnel111
L        192.168.1.1/32 is directly connected, Tunnel111
R1#
R1#
R1#
```

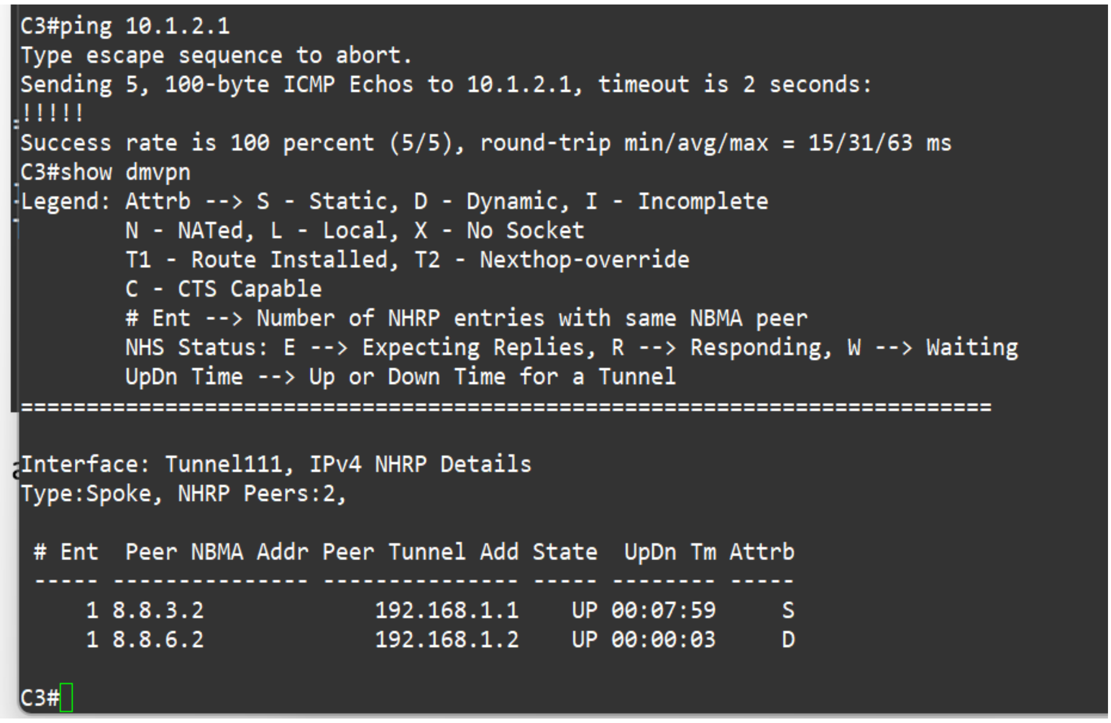
p. only for C3(C4 will be almost the same)

before ping:



```
C3#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel111, IPv4 NHRP Details
Type:Spoke, NHRP Peers:1,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 8.8.3.2             192.168.1.1    UP 00:00:49     S
```

after ping:



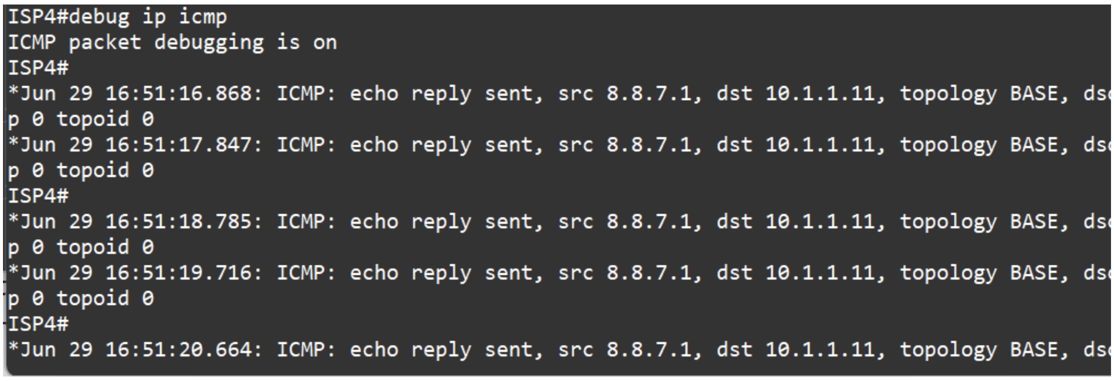
```
C3#ping 10.1.2.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.1.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 15/31/63 ms
C3#show dmvpn
Legend: Attrb --> S - Static, D - Dynamic, I - Incomplete
        N - NATed, L - Local, X - No Socket
        T1 - Route Installed, T2 - Nexthop-override
        C - CTS Capable
        # Ent --> Number of NHRP entries with same NBMA peer
        NHS Status: E --> Expecting Replies, R --> Responding, W --> Waiting
        UpDn Time --> Up or Down Time for a Tunnel
==========================================================================

Interface: Tunnel111, IPv4 NHRP Details
Type:Spoke, NHRP Peers:2,

 # Ent  Peer NBMA Addr Peer Tunnel Add State  UpDn Tm Attrb
 ----- --------------- --------------- ----- -------- -----
     1 8.8.3.2             192.168.1.1    UP 00:07:59     S
     1 8.8.6.2             192.168.1.2    UP 00:00:03     D

C3#
```

q. NO, the tunnel between C4 and C3 does not work and that is because we didn't establish a neighbor relationship on ISP1 to ISP3. That is a BGP problem.

s. the BGP configuration is important for the tunnel to work because we are trying the create a Full mesh DMVPN tunnels and in a full mesh DMVPN, each site (spoke) can directly communicate with every other site, bypassing the hub for direct spoke-to-spoke communication. This setup reduces the data path distance, potentially decreasing latency and improving bandwidth utilization. BGP is vital in managing the routes between these numerous endpoints efficiently.

8.NAT:

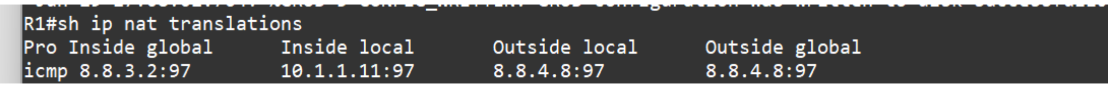
a. we can't ping from ubuntu1 to 8.8.7.1 and that is because NAT (Network Address Translation) is not enable on C1. Which mean that even tough ubuntu1 can send the traffic to C3 and C4 it does not how to send the traffic back again.



```
ISP4#debug ip icmp
ICMP packet debugging is on
ISP4#
*Jun 29 16:51:16.868: ICMP: echo reply sent, src 8.8.7.1, dst 10.1.1.11, topology BASE, ds
p 0 topoid 0
*Jun 29 16:51:17.847: ICMP: echo reply sent, src 8.8.7.1, dst 10.1.1.11, topology BASE, ds
p 0 topoid 0
ISP4#
*Jun 29 16:51:18.785: ICMP: echo reply sent, src 8.8.7.1, dst 10.1.1.11, topology BASE, ds
p 0 topoid 0
*Jun 29 16:51:19.716: ICMP: echo reply sent, src 8.8.7.1, dst 10.1.1.11, topology BASE, ds
p 0 topoid 0
ISP4#
*Jun 29 16:51:20.664: ICMP: echo reply sent, src 8.8.7.1, dst 10.1.1.11, topology BASE, ds
```

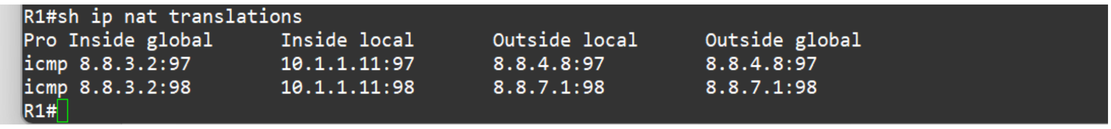
we can see that the traffic is arriving at ISP4 but the router dosnt have a route to network 10.X so if we will fix it we will be able to send ping from ubuntu1 to 8.8.7.1 successfully.

c.yes no it is working:



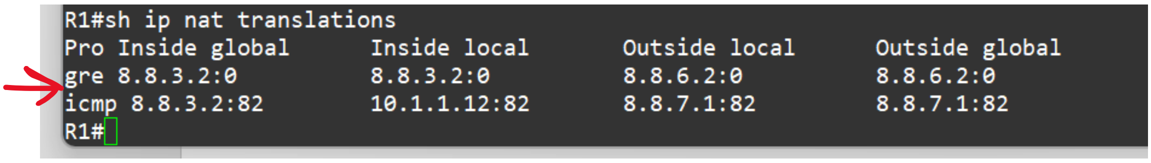
```
R1#sh ip nat translations
Pro Inside global       Inside local        Outside local       Outside global
icmp 8.8.3.2:97         10.1.1.11:97        8.8.4.8:97          8.8.4.8:97
```

d.



```
R1#sh ip nat translations
Pro Inside global       Inside local        Outside local       Outside global
icmp 8.8.3.2:97         10.1.1.11:97        8.8.4.8:97          8.8.4.8:97
icmp 8.8.3.2:98         10.1.1.11:98        8.8.7.1:98          8.8.7.1:98
R1#
```

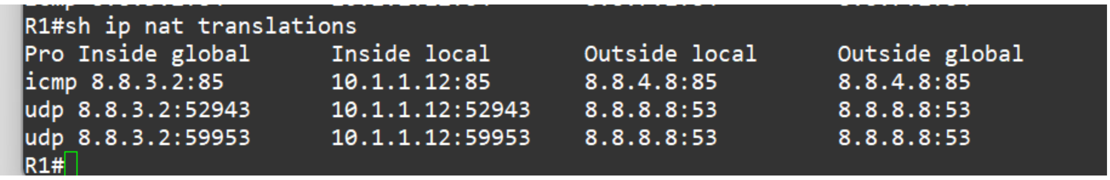
e.



```
R1#sh ip nat translations
Pro Inside global       Inside local        Outside local       Outside global
gre 8.8.3.2:0           8.8.3.2:0           8.8.6.2:0           8.8.6.2:0
icmp 8.8.3.2:82         10.1.1.12:82        8.8.7.1:82          8.8.7.1:82
R1#
```

Yes we can see that the GRE appears on the table.

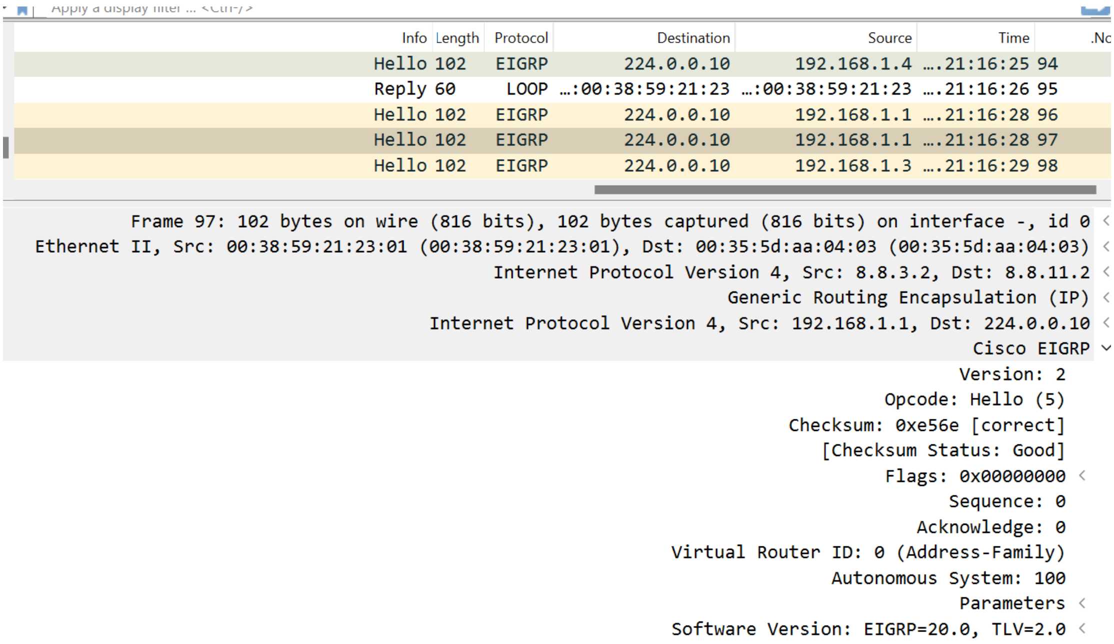
f. we can see that after senfing a ping fron ubuntu1 to cisco.com that the " show ip nat translations" table has changed.



```
R1#sh ip nat translations
Pro Inside global       Inside local        Outside local       Outside global
icmp 8.8.3.2:85         10.1.1.12:85        8.8.4.8:85          8.8.4.8:85
udp 8.8.3.2:52943       10.1.1.12:52943     8.8.8.8:53          8.8.8.8:53
udp 8.8.3.2:59953       10.1.1.12:59953     8.8.8.8:53          8.8.8.8:53
R1#
```

9. IPSEC (DMVPN security):

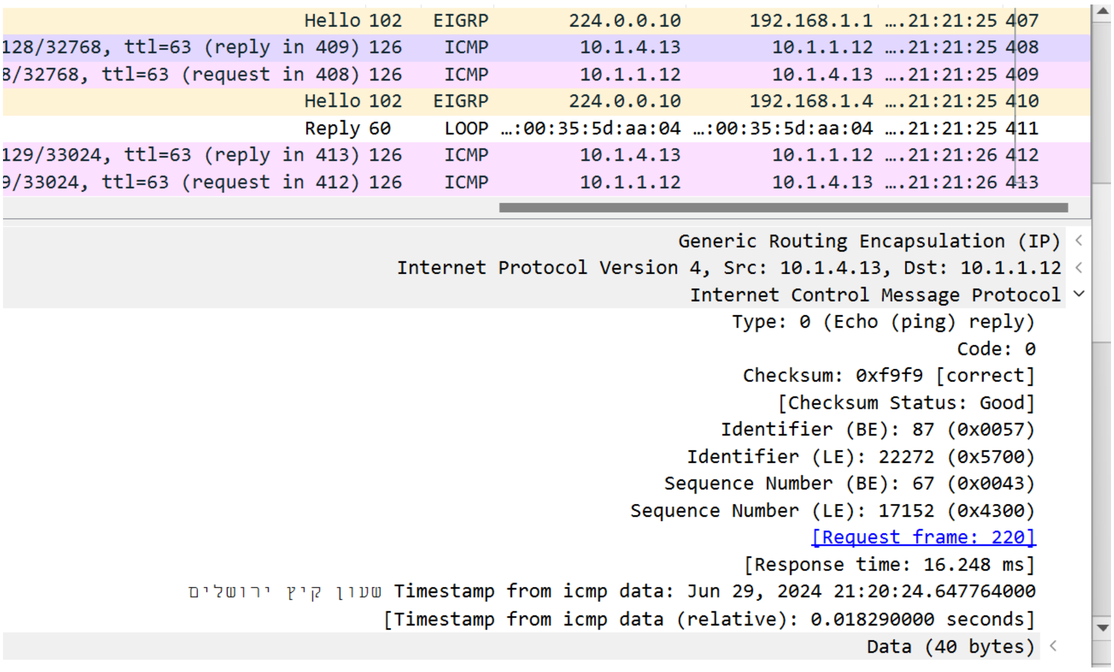
a.



```
        Info Length  Protocol       Destination             Source       Time   .No
      Hello 102      EIGRP           224.0.0.10          192.168.1.4  ….21:16:25 94
      Reply 60        LOOP  …:00:38:59:21:23  …:00:38:59:21:23  ….21:16:26 95
      Hello 102      EIGRP           224.0.0.10          192.168.1.1  ….21:16:28 96
      Hello 102      EIGRP           224.0.0.10          192.168.1.1  ….21:16:28 97
      Hello 102      EIGRP           224.0.0.10          192.168.1.3  ….21:16:29 98

         Frame 97: 102 bytes on wire (816 bits), 102 bytes captured (816 bits) on interface -, id 0 <
 Ethernet II, Src: 00:38:59:21:23:01 (00:38:59:21:23:01), Dst: 00:35:5d:aa:04:03 (00:35:5d:aa:04:03) <
                                       Internet Protocol Version 4, Src: 8.8.3.2, Dst: 8.8.11.2 <
                                                         Generic Routing Encapsulation (IP) <
                                  Internet Protocol Version 4, Src: 192.168.1.1, Dst: 224.0.0.10 <
                                                                                 Cisco EIGRP v
                                                                               Version: 2
                                                                         Opcode: Hello (5)
                                                              Checksum: 0xe56e [correct]
                                                                 [Checksum Status: Good]
                                                                    Flags: 0x00000000 <
                                                                           Sequence: 0
                                                                        Acknowledge: 0
                                                  Virtual Router ID: 0 (Address-Family)
                                                                Autonomous System: 100
                                                                           Parameters <
                                                  Software Version: EIGRP=20.0, TLV=2.0 <
```

as we can see from th EIGRP packet in the picture above we can read all the information inside the packet.

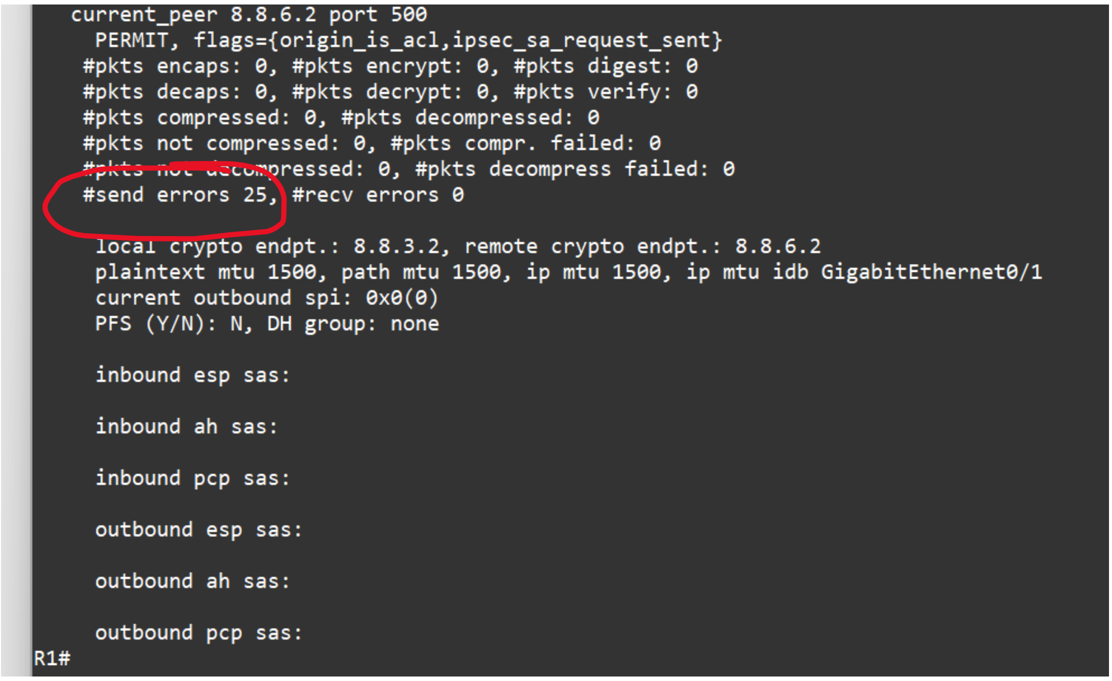
b. yes, the same as for the EIGRP packets.



```
                          Hello 102   EIGRP        224.0.0.10       192.168.1.1 ….21:21:25 407
128/32768, ttl=63 (reply in 409) 126   ICMP         10.1.4.13         10.1.1.12 ….21:21:25 408
3/32768, ttl=63 (request in 408) 126   ICMP         10.1.1.12         10.1.4.13 ….21:21:25 409
                          Hello 102   EIGRP        224.0.0.10       192.168.1.4 ….21:21:25 410
                          Reply 60     LOOP …:00:35:5d:aa:04 …:00:35:5d:aa:04 ….21:21:25 411
129/33024, ttl=63 (reply in 413) 126   ICMP         10.1.4.13         10.1.1.12 ….21:21:26 412
9/33024, ttl=63 (request in 412) 126   ICMP         10.1.1.12         10.1.4.13 ….21:21:26 413

                                                       Generic Routing Encapsulation (IP)
                               Internet Protocol Version 4, Src: 10.1.4.13, Dst: 10.1.1.12
                                                        Internet Control Message Protocol
                                                            Type: 0 (Echo (ping) reply)
                                                                                Code: 0
                                                             Checksum: 0xf9f9 [correct]
                                                               [Checksum Status: Good]
                                                         Identifier (BE): 87 (0x0057)
                                                      Identifier (LE): 22272 (0x5700)
                                                    Sequence Number (BE): 67 (0x0043)
                                                 Sequence Number (LE): 17152 (0x4300)
                                                                  [Request frame: 220]
                                                              [Response time: 16.248 ms]
            שעון קיץ ירושלים Timestamp from icmp data: Jun 29, 2024 21:20:24.647764000
                            [Timestamp from icmp data (relative): 0.018290000 seconds]
                                                                        Data (40 bytes)
```

d. there are no EIGRP neighbors anymore because we created an encrypted tunnel now and all the information is no longer readable via the network only in the end point.

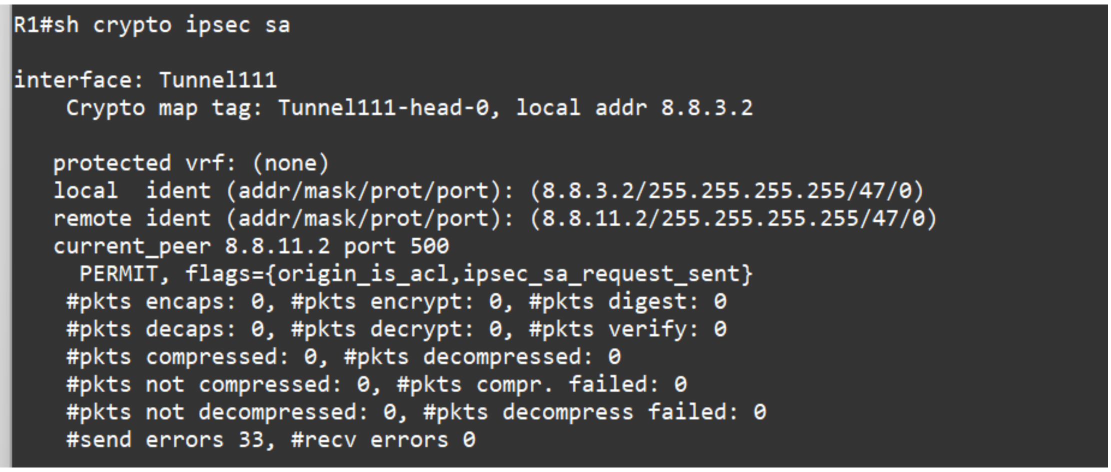
e. from C2:



```
    current_peer 8.8.6.2 port 500
      PERMIT, flags={origin_is_acl,ipsec_sa_request_sent}
     #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
     #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
     #pkts compressed: 0, #pkts decompressed: 0
     #pkts not compressed: 0, #pkts compr. failed: 0
     #pkts not decompressed: 0, #pkts decompress failed: 0
     #send errors 25, #recv errors 0

      local crypto endpt.: 8.8.3.2, remote crypto endpt.: 8.8.6.2
      plaintext mtu 1500, path mtu 1500, ip mtu 1500, ip mtu idb GigabitEthernet0/1
      current outbound spi: 0x0(0)
      PFS (Y/N): N, DH group: none

      inbound esp sas:

      inbound ah sas:

      inbound pcp sas:

      outbound esp sas:

      outbound ah sas:

      outbound pcp sas:
R1#
```

25 packets.

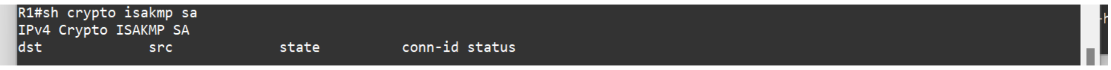
C4:



```
R1#sh crypto ipsec sa

interface: Tunnel111
    Crypto map tag: Tunnel111-head-0, local addr 8.8.3.2

   protected vrf: (none)
   local  ident (addr/mask/prot/port): (8.8.3.2/255.255.255.255/47/0)
   remote ident (addr/mask/prot/port): (8.8.11.2/255.255.255.255/47/0)
   current_peer 8.8.11.2 port 500
     PERMIT, flags={origin_is_acl,ipsec_sa_request_sent}
    #pkts encaps: 0, #pkts encrypt: 0, #pkts digest: 0
    #pkts decaps: 0, #pkts decrypt: 0, #pkts verify: 0
    #pkts compressed: 0, #pkts decompressed: 0
    #pkts not compressed: 0, #pkts compr. failed: 0
    #pkts not decompressed: 0, #pkts decompress failed: 0
    #send errors 33, #recv errors 0
```

33 packets.

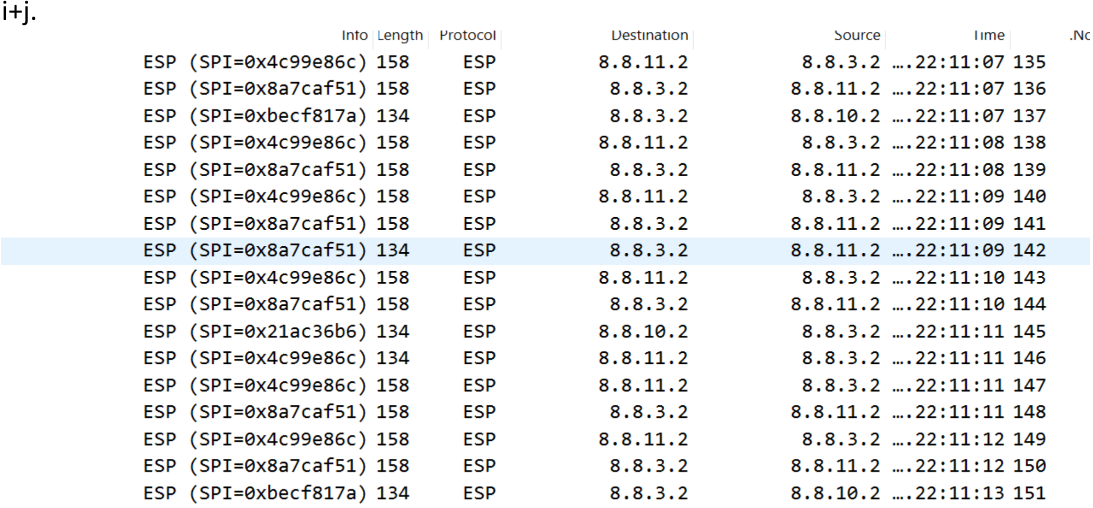
f.



```
R1#sh crypto isakmp sa
IPv4 Crypto ISAKMP SA
dst             src             state          conn-id status
```

there were deletions but I only manage to capture it in the end of the process

i+j.



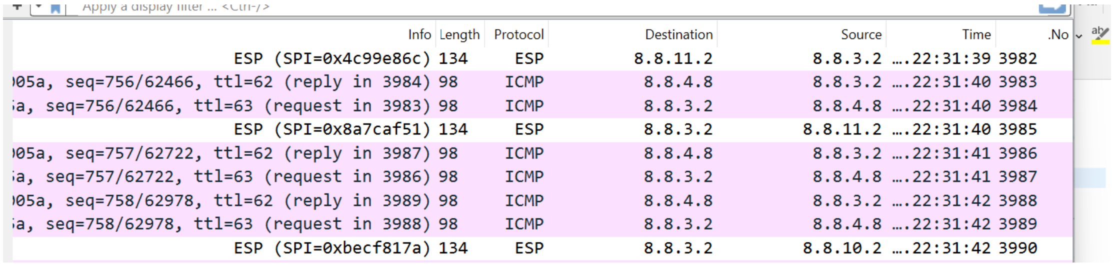
Info	Length	Protocol	Destination	Source	Time	.Nc
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:07	135
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:07	136
ESP (SPI=0xbecf817a)	134	ESP	8.8.3.2	8.8.10.2	….22:11:07	137
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:08	138
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:08	139
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:09	140
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:09	141
ESP (SPI=0x8a7caf51)	134	ESP	8.8.3.2	8.8.11.2	….22:11:09	142
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:10	143
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:10	144
ESP (SPI=0x21ac36b6)	134	ESP	8.8.10.2	8.8.3.2	….22:11:11	145
ESP (SPI=0x4c99e86c)	134	ESP	8.8.11.2	8.8.3.2	….22:11:11	146
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:11	147
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:11	148
ESP (SPI=0x4c99e86c)	158	ESP	8.8.11.2	8.8.3.2	….22:11:12	149
ESP (SPI=0x8a7caf51)	158	ESP	8.8.3.2	8.8.11.2	….22:11:12	150
ESP (SPI=0xbecf817a)	134	ESP	8.8.3.2	8.8.10.2	….22:11:13	151

As we can see from the capture there only ESP packets and there are no EIGRP or ICMP packets now.

When you encrypt a GRE (Generic Routing Encapsulation) tunnel, typically with IPsec for security, the original EIGRP (Enhanced Interior Gateway Routing Protocol) or ICMP (Internet Control Message Protocol) packets are encapsulated within the GRE packets, which are then further encapsulated within IPsec packets. The IPsec protocol primarily uses the ESP (Encapsulating Security Payload) packet format to provide confidentiality, integrity, and authenticity.

And that is why we can only find ESP packets.

k.



Info	Length	Protocol	Destination	Source	Time	.No
ESP (SPI=0x4c99e86c)	134	ESP	8.8.11.2	8.8.3.2	….22:31:39	3982
05a, seq=756/62466, ttl=62 (reply in 3984)	98	ICMP	8.8.4.8	8.8.3.2	….22:31:40	3983
a, seq=756/62466, ttl=63 (request in 3983)	98	ICMP	8.8.3.2	8.8.4.8	….22:31:40	3984
ESP (SPI=0x8a7caf51)	134	ESP	8.8.3.2	8.8.11.2	….22:31:40	3985
05a, seq=757/62722, ttl=62 (reply in 3987)	98	ICMP	8.8.4.8	8.8.3.2	….22:31:41	3986
a, seq=757/62722, ttl=63 (request in 3986)	98	ICMP	8.8.3.2	8.8.4.8	….22:31:41	3987
05a, seq=758/62978, ttl=62 (reply in 3989)	98	ICMP	8.8.4.8	8.8.3.2	….22:31:42	3988
a, seq=758/62978, ttl=63 (request in 3988)	98	ICMP	8.8.3.2	8.8.4.8	….22:31:42	3989
ESP (SPI=0xbecf817a)	134	ESP	8.8.3.2	8.8.10.2	….22:31:42	3990

As we can see, there are ICMP packets and that is because the route to cisco.com is not part of the VPN tunnel. So the traffic there is not encrypted.

Discussion

a. Yes, it is possible and can be done easily by configuring the Spoke routers to have more than one NHS address. Some reasons to do so include redundancy, load balancing, and supporting more than one HQ site.
b. #ip nhrp map multicast – This command enables the mGRE tunnel to forward multicast messages using OSPF, as it works with multicast addresses and is necessary..
c. To update each site's routing table with the proper entries (i.e., the site's LAN network), we need a dynamic routing protocol. This involves advertising the routes to the LAN networks through the mGRE tunnel interface and setting the IP next hop. Despite being a tunnel network, it can be viewed as an "overlay" network. Therefore, like any communication network, routing is essential, whether manually configured (static) or using a dynamic routing protocol. Static routes, as we know, are time-consuming and not scalable for large networks.
d. IPSec encrypted the encapsulated messages, including the mGRE part. The mGRE tunnel supports multicast messages ,but to achieve this, it encapsulates the multicast messages as unicast messages. IPSec has no problem encrypting these unicast messages.